# Course Project Documentation: Interview Simulator with A.I.

## Introduction

Motivation:

To improve user's interview skill to help user to prepare for their upcoming interview.

Task Description:

A web application using A.I. generated interview feedback to help improve student's interview skill.

Objective:

It provides the user with a common interview question and record their interview. Once the user finish recording their interview, the app will evaluate the interview by their body language and the tone of their voice.

## Problem Analysis

User Requirements

Since this is a web application, the user has to make sure that they have internet access and a web browser to search the website. Also, the user should check if their computer's camera and microphone are working properly before using the application.

User Goals

1. The website will first provide user with a common interview question.
2. Once the user read the question, they can click on a record button that will turn on their webcam and record the interview.
3. Once the user has completed the interview, the website will take user to the next page where it will analyze their interview. It will analyze user's body language and voice and gives a score from 0 to 100 in each quality. The quality includes eye visibility, sentiments, smiling, and looking at the camera. Then, it will take the average of all those qualities and show the final result to the user, and the process complete.
4. If the user wants to practice interview again, then they can go back to the home page to find another question to practice with and the process repeat.

Usability Requirements

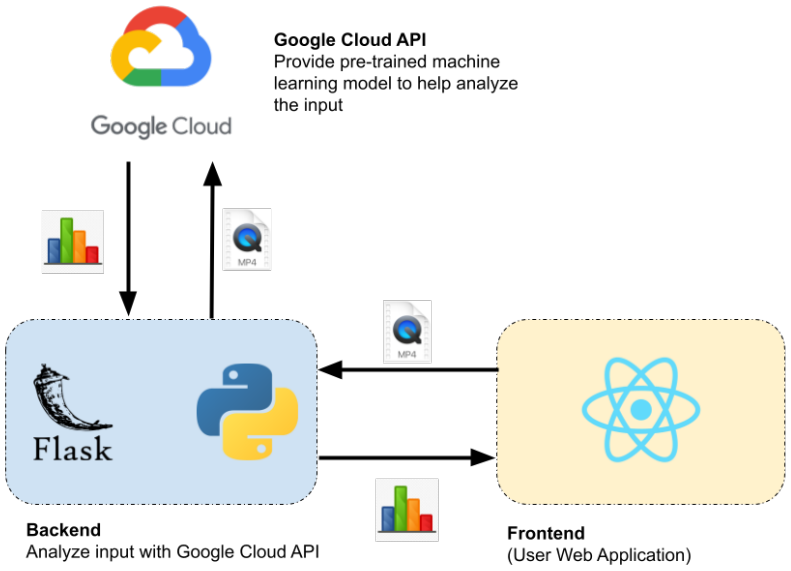
CLI

React frameworks[1] provides a CLI components that allows the developer to create frontend components and compile the code in order to present it on the browser.

API

Google Cloud API provides a pre-trained machine learning model, which include facial and speech recognition. These machine learning model allows the application to analyze user's body language and speech pattern when evaluating their interview. The benefit of having a pre-trained machine learning model is that it will speed up the process of our interview evaluation since it is time consuming to train machine learning model.

## Design




Architecture Overview (image above)

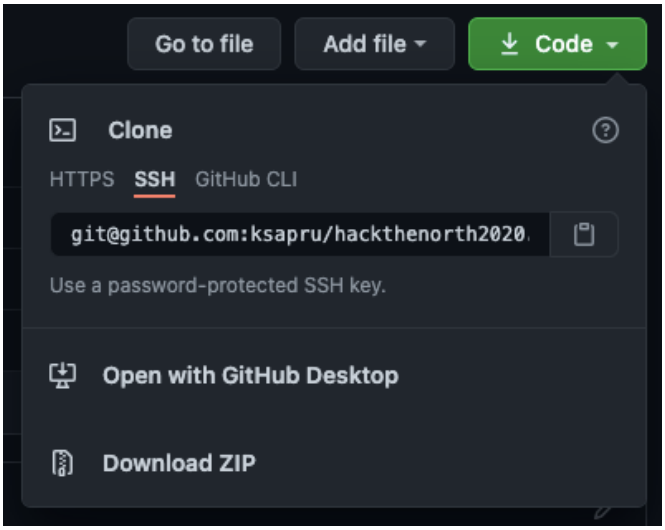
- The application has a frontend where the user records their interview, and a backend where it processes the user's recording for interview evaluation.
- When the user completed their interview, the website will save the recording as a video file (.mp4) and send it to the backend.
- As the backend receive the video recording, it will use an already trained machine learning model provide by Google Cloud API to analyze user's body language and speech patterns.
- Once the backend finished analyzing the recorded interview, it will send the score back to frontend, and it will present the score to the user.

## Installation

The application is stored as a repository in a code hosting platform called GitHub, which it's a website that allows developer to share their source code online. It also allows developers to collaborate with each other from anywhere around the world. The repository can be installed in any operating system (Windows, MacOS, Linux, and etc).

To install the Source Code:

1. Make sure the dependencies are installed before installing the actual source code, which are React[1], GIT[2], NodeJS[3], and Python[4]
2. Install in two ways
    a. Go to the repository link on GitHub
        i. Click on the button Code on the right-hand side
        ii. Click on Download ZIP to install the application
    b. Open the terminal and clone the repository
        i. Command: git clone <repository-link>
        ii. Note: the user can find the repository link after clicking the button Code on GitHub




Note: The application is only accessible with the teammate so far but will be release public in the end of April.

**Project Management**

Software Development

Programming Language(s) Used

- Python is used for creating backend server with the help of its backend framework Flask. It is the language that allows the developer to access functionality from Google Cloud API.
- JavaScript is used for creating user interface with the frontend framework React.

Development Platform

Visual Studio Code (VS Code) is the development platform that the team use for developing this application. VS Code can be installed in any operating system (Windows, MacOS, and Linux), and it supports a variety of programming languages and web frameworks. In VS Code, it has a built-in terminal that allows developer to compile their code through the editor.

Version Control System

The team use GIT as the version control system, which it tracks changes of the project from each developer. Also, since GIT is integrated with the website GitHub, this allows the team to collaborate during quarantine.

Project Organization

Process Model

The team use the agile model where the requirements and solution of the application will evolve through each iteration. The team will have their short daily stand-up meeting where they discuss the process of their work and address problems encounter along the way. This method allows the team to quickly spot problems and fix it right away.

Project Responsibilities

- Developer 1 and 2: Create prototype design of the frontend and implement it with React.
- Developer 3 and 4: Create backend that can integrate with Google Cloud API to allow the backend to analyze recorded interview from the user.
- Developer 5: Supervise developers from the development of both frontend and backend, and integrate both frontend and backend to make sure that data can transfer smoothly between them.

Project Deliverables

- This application will be available and accessible in any browsers that the user preferred.
- The application will provide user with common interview questions.
- User can practice their interview with this application as many times as possible.
- User will get A.I. generated feedback to improve their interview skills.

Milestones and Schedule

| | |
|---|---|
| February 12 | Recruit people to create a group |
| February 19 | Pick a topic |
| March 5 | Develop backend |
| March 9 | Integrate backend with Google Cloud API |
| March 10 | Complete unit test for backend |
| March 19 | Design and implement frontend |
| March 26 | Complete unit test for backend |
| March 28 | Finalize application and write more test |
| April 5 | Project deadline |

Risk Management

The team has very little experience with React and Google Cloud API, it is the first time that the team have work with these two technologies. Since there are two developers (developer 1 and 2) who have experience working with JavaScript, they will focus on the frontend and learn React. This allows the other two developers (developer 3, 4 and 5) to focus on the backend development and research on integrating Google Cloud API with it. One of the developers (developer 5) who worked as a full stack developer on his previous Co-op, he will supervise on the development of both the frontend and backend. Since everyone is still in quarantine, the team use an app called Discord to communicate with each other. Discord also provides screen share during calls to allow developer to show off their development progress. Each developer will work on the task that they are assigned. If one developer encountered a difficult problem that he or she cannot solve, then they can address it at the stand-up meeting, and developer 5 will arrange an available time to have a call to solve the problem.

**References:**

[1] Codecademy, “React Setup, Part I: React and ReactDOM” https://www.codecademy.com/articles/react-setup-i (access March 23, 2021)

[2] GIT, “Getting Started Installing Git” https://git-scm.com/book/en/v2/Getting-Started-Installing-Git (access March 23, 2021)

[3] Phoenixnap, “How To Install Node.Js And NPM On Windows” https://phoenixnap.com/kb/install-node-js-npm-on-windows (posted October 28, 2019)

[4] Phoenixnap, “How To Install Python 3 On Windows 10” https://phoenixnap.com/kb/how-to-install-python-3-windows (posted April 2, 2019)